Refine Search

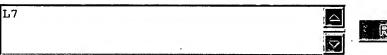## Search Results -

Terms	Documents
L3 and L6	135

Database:

US Pre-Grant Publication Full-Text Database
US Patents Full-Text Database
US OCR Full-Text Database
EPO Abstracts Database
JPO Abstracts Database
Derwent World Patents Index
IBM Technical Disclosure Bulletins

Search:

L7

Refine Search

Recall Text   Clear   Interrupt

Search History

DATE: Wednesday, August 10, 2005   Printable Copy   Create Case

Set Name side by side	Query	Hit Count	Set Name result set
DB=PGPB,USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=NO; OP=ADJ			
L7	l3 and L6	135	L7
L6	l4 same L5	451	L6
L5	prefetch$4 or pre-fetch$4 or look$4-ahead or lookahead or (look$4 adj ahead)	20339	L5
L4	(effect or impact or result) with (request or transaction or command)	81118	L4
L3	l1 or L2	2902	L3
L2	712/205,207,233-241.ccls.	2248	L2
L1	711/137.ccls.	816	L1

END OF SEARCH HISTORY

Refine Search

Search Results -
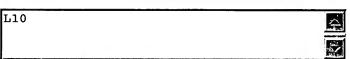
Terms	Documents
L8 and L4	44

Database:

US Pre-Grant Publication Full-Text Database
US Patents Full-Text Database
US OCR Full-Text Database
EPO Abstracts Database
JPO Abstracts Database
Derwent World Patents Index
IBM Technical Disclosure Bulletins

Search:

L10

Refine Search

Recall Text   Clear   Interrupt

Search History

DATE: **Wednesday, August 10, 2005**   Printable Copy   Create Case

Set Name side by side	Query	Hit Count	Set Name result set
DB=PGPB,USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=NO; OP=ADJ			
L10	18 and 14	44	L10
L9	16 and L8	1	L9
L8	kurokawa.in.	10962	L8
L7	13 and L6	135	L7
L6	14 same L5	451	L6
L5	prefetch$4 or pre-fetch$4 or look$4-ahead or lookahead or (look$4 adj ahead)	20339	L5
L4	(effect or impact or result) with (request or transaction or command)	81118	L4
L3	11 or L2	2902	L3
L2	712/205,207,233-241.ccls.	2248	L2
L1	711/137.ccls.	816	L1

END OF SEARCH HISTORY

IEEE Xplore RELEASE 2.1

Welcome United States Patent and Trademark Office

□ Search Results   BROWSE   SEARCH   IEEE XPLORE GUIDE

Results for "(((impact<or>effect)<and>prefetch*)<in>metadata)"   ☑ e-mail
Your search matched 83 of 1222090 documents.
A maximum of 100 results are displayed, 25 to a page, sorted by **Relevance** in **Descending** order.

» Search Options

View Session History

New Search

» Key

IEEE JNL	IEEE Journal or Magazine
IEE JNL	IEE Journal or Magazine
IEEE CNF	IEEE Conference Proceeding
IEE CNF	IEE Conference Proceeding
IEEE STD	IEEE Standard

Modify Search

(((impact<or>effect)<and>prefetch*)<in>metadata)   ≫

☐ Check to search only within this results set

Display Format: ⦿ Citation   ○ Citation & Abstract

| Select | Article Information | View: 1-25 |

☐ 1. **A prefetch taxonomy**
Viji Srinivasan; Davidson, E.S.; Tyson, G.S.;
Computers, IEEE Transactions on
Volume 53, Issue 2, Feb 2004 Page(s):126 - 140
Digital Object Identifier 10.1109/TC.2004.1261824

AbstractPlus | Full Text: PDF(1617 KB)   IEEE JNL

☐ 2. **Bundling: reducing the overhead of multiprocessor prefetchers**
Wallin, D.; Hagersten, E.;
Parallel and Distributed Processing Symposium, 2004. Proceedings. 18th International
26-30 April 2004 Page(s):74
Digital Object Identifier 10.1109/IPDPS.2004.1303006

AbstractPlus | Full Text: PDF(2404 KB)   IEEE CNF

☐ 3. **Cost-effective compiler directed memory prefetching and bypassing**
Ortega, D.; Ayguade, E.; Baer, J.-L.; Valero, M.;
Parallel Architectures and Compilation Techniques, 2002. Proceedings. 2002 International Confere
22-25 Sept. 2002 Page(s):189 - 198
Digital Object Identifier 10.1109/PACT.2002.1106017

AbstractPlus | Full Text: PDF(1356 KB)   IEEE CNF

☐ 4. **Multi-level memory prefetching for media and stream processing**
Fritts, J.;
Multimedia and Expo, 2002. ICME '02. Proceedings. 2002 IEEE International Conference on
Volume 2, 26-29 Aug. 2002 Page(s):101 - 104 vol.2
Digital Object Identifier 10.1109/ICME.2002.1035522

AbstractPlus | Full Text: PDF(594 KB)   IEEE CNF

☐ 5. **The network effects of prefetching**
Crovella, M.; Barford, P.;
INFOCOM '98. Seventeenth Annual Joint Conference of the IEEE Computer and Communications
Proceedings. IEEE
Volume 3, 29 March-2 April 1998 Page(s):1232 - 1239 vol.3
Digital Object Identifier 10.1109/INFCOM.1998.662937

AbstractPlus | Full Text: PDF(932 KB)   IEEE CNF

☐ 6. **The impact of parallel loop scheduling strategies on prefetching in a shared memory multip**

Lilja, D.J.;
Parallel and Distributed Systems, IEEE Transactions on
Volume 5, Issue 6, June 1994 Page(s):573 - 584
Digital Object Identifier 10.1109/71.285604

AbstractPlus | Full Text: PDF(1288 KB)   IEEE JNL

7. **A keyword-based semantic prefetching approach in Internet news services**
Cheng-Zhong Xu; Ibrahim, T.I.;
Knowledge and Data Engineering, IEEE Transactions on
Volume 16, Issue 5, May 2004 Page(s):601 - 611
Digital Object Identifier 10.1109/TKDE.2004.1277820

AbstractPlus | Full Text: PDF(996 KB)   IEEE JNL

8. **Prefetching scheme for image processing on shared memory multiprocessors**
Rhee, Y.; Lee, J.;
Image Processing, 1996. Proceedings., International Conference on
Volume 1, 16-19 Sept. 1996 Page(s):157 - 160 vol.2
Digital Object Identifier 10.1109/ICIP.1996.560626

AbstractPlus | Full Text: PDF(340 KB)   IEEE CNF

9. **Pointer cache assisted prefetching**
Collins, J.; Sair, S.; Calder, B.; Tullsen, D.M.;
Microarchitecture, 2002. (MICRO-35). Proceedings. 35th Annual IEEE/ACM International Symposii
18-22 Nov. 2002 Page(s):62 - 73
Digital Object Identifier 10.1109/MICRO.2002.1176239

AbstractPlus | Full Text: PDF(271 KB)   IEEE CNF

10. **Sequential hardware prefetching in shared-memory multiprocessors**
Dahlgren, F.; Dubois, M.; Stenstrom, P.;
Parallel and Distributed Systems, IEEE Transactions on
Volume 6, Issue 7, July 1995 Page(s):733 - 746
Digital Object Identifier 10.1109/71.395402

AbstractPlus | References | Full Text: PDF(1444 KB)   IEEE JNL

11. **Integrating Web caching and Web prefetching in client-side proxies**
Wei-Guang Teng; Cheng-Yue Chang; Ming-Syan Chen;
Parallel and Distributed Systems, IEEE Transactions on
Volume 16, Issue 5, May 2005 Page(s):444 - 455
Digital Object Identifier 10.1109/TPDS.2005.56

AbstractPlus | Full Text: PDF(880 KB)   IEEE JNL

12. **Webspace surfing patterns and their impact on Web prefetching**
Khan, J.I.; Qingping Tao;
Cyberworlds, 2003. Proceedings. 2003 International Conference on
2003 Page(s):478 - 485
Digital Object Identifier 10.1109/CYBER.2003.1253493

AbstractPlus | Full Text: PDF(282 KB)   IEEE CNF

13. **Impact of Compiler-based Data-Prefetching Techniques on SPEC OMP Application Perform:**
Xinmin Tian; Krishnaiyer, R.; Saito, H.; Girkar, M.; Wei Li;
Parallel and Distributed Processing Symposium, 2005. Proceedings. 19th IEEE International
04-08 April 2005 Page(s):53a - 53a
Digital Object Identifier 10.1109/IPDPS.2005.248

AbstractPlus | Full Text: PDF(152 KB)   IEEE CNF

14. **The effect of limited network bandwidth and its utilization by latency hiding techniques in la memory systems**
Sunil Kim; Veidenbaum, A.V.;

Parallel Architectures and Compilation Techniques., 1997. Proceedings. 1997 International Conferen
10-14 Nov. 1997 Page(s):40 - 51
Digital Object Identifier 10.1109/PACT.1997.644002

AbstractPlus | Full Text: PDF(1284 KB)   IEEE CNF

15. **Effective Instruction prefetching in chip multiprocessors for modern commercial applicatior**
Spracklen, L.; Yuan Chou; Abraham, S.G.;
High-Performance Computer Architecture, 2005. HPCA-11. 11th International Symposium on
12-16 Feb. 2005 Page(s):225 - 236
Digital Object Identifier 10.1109/HPCA.2005.13

AbstractPlus | Full Text: PDF(192 KB)   IEEE CNF

16. **Differences in cost and benefit of prefetching in circuit-switched and packet-switched netwc**
Angermann, M.;
Telecommunications, 2003. ICT 2003. 10th International Conference on
Volume 2, 23 Feb.-1 March 2003 Page(s):1084 - 1090 vol.2
Digital Object Identifier 10.1109/ICTEL.2003.1191588

AbstractPlus | Full Text: PDF(558 KB)   IEEE CNF

17. **A decoupled predictor-directed stream prefetching architecture**
Sair, S.; Sherwood, T.; Calder, B.;
Computers, IEEE Transactions on
Volume 52, Issue 3, March 2003 Page(s):260 - 276
Digital Object Identifier 10.1109/TC.2003.1183943

AbstractPlus | References | Full Text: PDF(1356 KB)   IEEE JNL

18. **Effect of speculative prefetching on network load in distributed systems**
Tuah, N.J.; Kumar, M.; Venkatesh, S.;
Parallel and Distributed Processing Symposium., Proceedings 15th International
23-27 April 2001 Page(s):6 pp.
Digital Object Identifier 10.1109/IPDPS.2001.924979

AbstractPlus | Full Text: PDF(200 KB) IEEE CNF

19. **A study of measurement-based Web prefetch control**
Lei Wang; Lianfang Zhang; Yantai Shu; Miao Dong; Yang, O.W.W.;
Electrical and Computer Engineering, 2000 Canadian Conference on
Volume 1, 7-10 March 2000 Page(s):204 - 208 vol.1
Digital Object Identifier 10.1109/CCECE.2000.849699

AbstractPlus | Full Text: PDF(408 KB)   IEEE CNF

20. **Multimedia prefetching strategy for news-on-demand applications**
Mostefaoui, A.; Brunie, L.;
Database and Expert Systems Applications, 1999. Proceedings. Tenth International Workshop on
1-3 Sept. 1999 Page(s):24 - 28
Digital Object Identifier 10.1109/DEXA.1999.795119

AbstractPlus | Full Text: PDF(76 KB)   IEEE CNF

21. **Tolerating medium latencies on data caches with hardware-based prefetching**
Moreno, E.D.; Kofuji, S.T.; Martins, C.A.P.S.;
Electrical and Computer Engineering, 1997. IEEE 1997 Canadian Conference on
Volume 2, 25-28 May 1997 Page(s):720 - 723 vol.2
Digital Object Identifier 10.1109/CCECE.1997.608340

AbstractPlus | Full Text: PDF(324 KB)   IEEE CNF

22. **Predictor-directed stream buffers**
Sherwood, T.; Sair, S.; Calder, B.;
Microarchitecture, 2000. MICRO-33. Proceedings. 33rd Annual IEEE/ACM International Symposiur
10-13 Dec. 2000 Page(s):42 - 53

Digital Object Identifier 10.1109/MICRO.2000.898057

AbstractPlus | Full Text: PDF(1040 KB)   IEEE CNF

23. **A comparison of architectural support for messaging in the TMC CM-5 and the Cray T3D**
Karamcheti, V.; Chien, A.A.;
Computer Architecture, 1995. Proceedings. 22nd Annual International Symposium on
22-24 Jun 1995 Page(s):298 - 307

AbstractPlus | Full Text: PDF(1012 KB)   IEEE CNF

24. **Effective Instruction Prefetching via Fetch Prestaging**
Falcon, A.; Ramirez, A.; Valero, M.;
Parallel and Distributed Processing Symposium, 2005. Proceedings. 19th IEEE International
04-08 April 2005 Page(s):20b - 20b
Digital Object Identifier 10.1109/IPDPS.2005.188

AbstractPlus | Full Text: PDF(216 KB)   IEEE CNF

25. **Prefetching the means for document transfer: a new approach for reducing Web latency**
Cohen, E.; Kaplan, H.;
INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications
Proceedings. IEEE
Volume 2, 26-30 March 2000 Page(s):854 - 863 vol.2
Digital Object Identifier 10.1109/INFCOM.2000.832260

AbstractPlus | Full Text: PDF(1112 KB)   IEEE CNF

View: **1-25** |

Help   Contact Us   Privac

© Copyright 2005 IE

Indexed by
Ⅱ Inspec˙

P⊚RTAL
USPTO

Search: ⊂ The ACM Digital Library   ⊙ The Guide

effect +prefetch*

SEARCH

THE GUIDE TO COMPUTING LITERATURE

Feedback Report a problem Satisfaction survey

Terms used **effect prefetch**   Found **3,033** of 873,144

Sort results by	relevance ▽	◆ Save results to a Binder
Display results	expanded form ▽	? Search Tips
		☐ Open results in a new window

Try an Advanced Search
Try this search in The Digital Library

Results 1 - 20 of 200   Result page: **1** 2 3 4 5 6 7 8 9 10   next
Best 200 shown   Relevance scale ☐ ▭ ▬ ◼ ■

1 Effective cache prefetching on bus-based multiprocessors   ■
Dean M. Tullsen, Susan J. Eggers
February 1995 **ACM Transactions on Computer Systems (TOCS)**, Volume 13 Issue 1

Full text available: 🅰 pdf(2.30 MB)   Additional Information: full citation, abstract, references, citings, index terms

Compiler-directed cache prefetching has the potential to hide much of the high memory latency seen by current and future high-performance processors. However, prefetching is not without costs, particularly on a shared-memory multiprocessor. Prefetching can negatively affect bus utilization, overall cache miss rates, memory latencies and data sharing. We simulate the effects of a compiler-directed prefetching algorithm, running on a range of bus-based multiprocessors. We show that, despite a ...

Keywords: bus-based multiprocessors, cache prefetching, false sharing, memory latency hiding

2 Architectural and compiler support for effective instruction prefetching: a cooperative approach   ■
February 2001 **ACM Transactions on Computer Systems (TOCS)**, Volume 19 Issue 1

Full text available: 🅰 pdf(432.96 KB)   Additional Information: full citation, abstract, references, citings, index terms, review

Instruction cache miss latency is becoming an increasingly important performance bottleneck, especially for commercial applications. Although instruction prefetching is an attractive technique for tolerating this latency, we find that existing prefetching schemes are insufficient for modern superscalar processors, since they fail to issue prefetches early enough (particularly for nonsequential accesses). To overcome these limitations, we propose a new instruction prefetching technique where ...

Keywords: compiler optimization, instruction prefetching

3 Effective jump-pointer prefetching for linked data structures   ■
Amir Roth, Gurindar S. Sohi
May 1999 **ACM SIGARCH Computer Architecture News , Proceedings of the 26th annual international symposium on Computer architecture**, Volume 27 Issue 2

Full text available: 🅰 pdf(113.33 KB)   Additional Information:

Publisher Site

Current techniques for prefetching linked data structures (LDS) exploit the work available in one loop iteration or recursive call to overlap pointer chasing latency. Jump pointers, which provide direct access to non-adjacent nodes, can be used for prefetching when loop and recursive procedure bodies are small and do not have sufficient work to overlap a long latency. This paper describes a framework for jump-pointer prefetching (JPP) that supports four prefetching idioms: queue, full, chain, an ...

4 Simple and effective array prefetching in Java
Brendon Cahoon, Kathryn S. McKinley
November 2002 **Proceedings of the 2002 joint ACM-ISCOPE conference on Java Grande**

Full text available: pdf(118.49 KB)   Additional Information: full citation, abstract, references, citings, index terms

Java is becoming a viable choice for numerical algorithms due to the software engineering benefits of object-oriented programming. Because these programs still use large arrays that do not fit in the cache, they continue to suffer from poor memory performance. To hide memory latency, we describe a new unified compile-time analysis for software prefetching arrays and linked structures in Java. Our previous work uses data-flow analysis to discover linked data structure accesses, and here we presen ...

Keywords: Java, array prefetching, memory optimization, static analysis

5 Cooperative prefetching: compiler and hardware support for effective instruction prefetching in modern processors
Chi-Keung Luk, Todd C. Mowry
November 1998 **Proceedings of the 31st annual ACM/IEEE international symposium on Microarchitecture**
Full text available: pdf(3.41 MB)   Additional Information: full citation, references, citings, index terms

6 An effective programmable prefetch engine for on-chip caches
Tien-Fu Chen
December 1995 **Proceedings of the 28th annual international symposium on Microarchitecture**
Full text available: pdf(721.83 KB)   Additional Information: full citation, references, citings, index terms

7 Data prefetch mechanisms
Steven P. Vanderwiel, David J. Lilja
June 2000 **ACM Computing Surveys (CSUR)**, Volume 32 Issue 2

Full text available: pdf(172.07 KB)   Additional Information: full citation, abstract, references, citings, index terms, review

The expanding gap between microprocessor and DRAM performance has necessitated the use of increasingly aggressive techniques designed to reduce or hide the latency of main memory access. Although large cache hierarchies have proven to be effective in reducing this latency for the most frequently used data, it is still not uncommon for many programs to spend more than half their run times stalled on memory requests. Data prefetching has been proposed as a technique for hiding the access lat ...

Keywords: memory latency, prefetching

8 Speeding up irregular applications in shared-memory multiprocessors: memory binding and group prefetching

Zheng Zhang, Josep Torrellas

May 1995 **ACM SIGARCH Computer Architecture News , Proceedings of the 22nd annual international symposium on Computer architecture**, Volume 23 Issue 2

Full text available: pdf(1.74 MB)   Additional Information: full citation, abstract, references, citings, index terms

While many parallel applications exhibit good spatial locality, other important codes in areas like graph problem-solving or CAD do not. Often, these irregular codes contain small records accessed via pointers. Consequently, while the former applications benefit from long cache lines, the latter prefer short lines. One good solution is to combine short lines with prefetching. In this way, each application can exploit the amount of spatial locality that it has. However, prefetching, if provided, ...

9 Dependence based prefetching for linked data structures

Amir Roth, Andreas Moshovos, Gurindar S. Sohi

October 1998 **Proceedings of the eighth international conference on Architectural support for programming languages and operating systems**, Volume 32 , 33 Issue 5 , 11

Full text available: pdf(1.81 MB)   Additional Information: full citation, abstract, references, citings, index terms

We introduce a dynamic scheme that captures the accesspat-terns of linked data structures and can be used to predict future accesses with high accuracy. Our technique exploits the dependence relationships that exist between loads that produce addresses and loads that consume these addresses. By identzj+ing producer-consumer pairs, we construct a compact internal representation for the associated structure and its traversal. To achieve a prefetching eflect, a small prefetch engine speculatively t ...

10 Tolerating memory latency through push prefetching for pointer-intensive applications

Chia-Lin Yang, Alvin R. Lebeck, Hung-Wei Tseng, Chien-Hao Lee

December 2004 **ACM Transactions on Architecture and Code Optimization (TACO)**, Volume 1 Issue 4

Full text available: pdf(590.24 KB)   Additional Information: full citation, abstract, references, index terms

Prefetching is often used to overlap memory latency with computation for array-based applications. However, prefetching for pointer-intensive applications remains a challenge because of the irregular memory access pattern and pointer-chasing problem. In this paper, we proposed a cooperative hardware/software prefetching framework, the push architecture, which is designed specifically for linked data structures. The push architecture exploits program structure for future address generation instea ...

Keywords: Prefetch, linked data structures, memory hierarchy, pointer-chasing

11 Limitations of cache prefetching on a bus-based multiprocessor

Dean M. Tullsen, Susan J. Eggers

May 1993 **ACM SIGARCH Computer Architecture News , Proceedings of the 20th annual international symposium on Computer architecture**, Volume 21 Issue 2

Full text available: pdf(1.10 MB)   Additional Information: full citation, abstract, references, citings, index terms

Compiler-directed cache prefetching has the potential to hide much of the high memory latency seen by current and future high-performance processors. However, prefetching is not without costs, particularly on a multiprocessor. Prefetching can negatively affect bus

utilization, overall cache miss rates, memory latencies and data sharing. We simulated the effects of a particular compiler-directed prefetching algorithm, running on a bus-based multiprocesssor. We showed that, despite a high mem ...

12 Design and evaluation of a compiler algorithm for prefetching

Todd C. Mowry, Monica S. Lam, Anoop Gupta

September 1992 **ACM SIGPLAN Notices , Proceedings of the fifth international conference on Architectural support for programming languages and operating systems,** Volume 27 Issue 9

Full text available: pdf(1.70 MB)   Additional Information: full citation, references, citings, index terms

13 Tolerating latency in multiprocessors through compiler-inserted prefetching

Todd C. Mowry

February 1998 **ACM Transactions on Computer Systems (TOCS),** Volume 16 Issue 1

Full text available: pdf(410.70 KB)   Additional Information: full citation, abstract, references, citings, index terms, review

The large latency of memory accesses in large-scale shared-memory multiprocessors is a key obstacle to achieving high processor utilization. Software-controlled prefetching is a technique for tolerating memory latency by explicitly executing instructions to move data close to the processor before the data are actually needed. To minimize the burden on the programmer, compiler support is needed to automatically insert prefetch instructions into the code. A key challenge when ...

Keywords: compiler optimization, prefetching

14 Clustered microarchitectures: Cluster prefetch: tolerating on-chip wire delays in clustered microarchitectures

Rajeev Balasubramonian

June 2004 **Proceedings of the 18th annual international conference on Supercomputing**

Full text available: pdf(153.44 KB)   Additional Information: full citation, abstract, references, index terms

The growing dominance of wire delays at future technology points renders a microprocessor communication-bound. Clustered microarchitectures allow most dependence chains to execute without being affected by long on-chip wire latencies. They also allow faster clock speeds and reduce design complexity, thereby emerging as a popular design choice for future microprocessors. However, a centralized data cache threatens to be the primary bottle-neck in highly clustered systems. The paper attempts to id ...

Keywords: clustered microarchitectures, communication-bound processors, data prefetch, distributed caches, effective address and memory dependence prediction, processor

15 Software prefetching for mark-sweep garbage collection: hardware analysis and software redesign

Chen-Yong Cher, Antony L. Hosking, T. N. Vijaykumar

October 2004 **Proceedings of the 11th international conference on Architectural support for programming languages and operating systems,** Volume 38 , 39 , 32 Issue 5 , 11 , 5

Full text available: pdf(165.32 KB)   Additional Information: full citation, abstract, references, index terms

Tracing garbage collectors traverse references from live program variables, transitively tracing out the closure of live objects. Memory accesses incurred during tracing are

essentially random: a given object may contain references to any other object. Since application heaps are typically much larger than hardware caches, tracing results in many cache misses. Technology trends will make cache misses more important, so tracing is a prime target for prefetching.Simulation of Java benchmarks runni ...

Keywords: breadth-first, buffered prefetch, cache architecture, depth-first, garbage collection, mark-sweep, prefetch-on-grey, prefetching

16 The interaction of software prefetching with ILP processors in shared-memory systems

Parthasarathy Ranganathan, Vijay S. Pai, Hazim Abdel-Shafi, Sarita V. Adve
May 1997 **ACM SIGARCH Computer Architecture News , Proceedings of the 24th annual international symposium on Computer architecture**, Volume 25 Issue 2

Full text available: pdf(2.44 MB)   Additional Information: full citation, abstract, references, citings, index terms

Current microprocessors aggressively exploit instruction-level parallelism (ILP) through techniques such as multiple issue, dynamic scheduling, and non-blocking reads. Recent work has shown that memory latency remains a significant performance bottleneck for shared-memory multiprocessor systems built of such processors.This paper provides the first study of the effectiveness of software-controlled non-binding prefetching in shared memory multiprocessors built of state-of-the-art ILP-based proces ...

17 Software prefetching

David Callahan, Ken Kennedy, Allan Porterfield
April 1991 **Proceedings of the fourth international conference on Architectural support for programming languages and operating systems**, Volume 19 , 25 , 26 Issue 2 , Special Issue , 4

Full text available: pdf(1.25 MB)   Additional Information: full citation, references, citings, index terms

18 A general framework for prefetch scheduling in linked data structures and its application to multi-chain prefetching

Seungryul Choi, Nicholas Kohout, Sumit Pamnani, Dongkeun Kim, Donald Yeung
May 2004 **ACM Transactions on Computer Systems (TOCS)**, Volume 22 Issue 2

Full text available: pdf(2.45 MB)   Additional Information: full citation, abstract, references, index terms

Pointer-chasing applications tend to traverse composite data structures consisting of multiple independent pointer chains. While the traversal of any single pointer chain leads to the serialization of memory operations, the traversal of independent pointer chains provides a source of memory parallelism. This article investigates exploiting such *interchain memory parallelism* for the purpose of memory latency tolerance, using a technique called *multi-chain prefetching*. Previous work ...

Keywords: Data prefetching, memory parallelism, pointer-chasing code

19 Call graph prefetching for database applications

Murali Annavaram, Jignesh M. Patel, Edward S. Davidson
November 2003 **ACM Transactions on Computer Systems (TOCS)**, Volume 21 Issue 4

Full text available: pdf(701.71 KB)   Additional Information: full citation, abstract, references, index terms

With the continuing technological trend of ever cheaper and larger memory, most data sets in database servers will soon be able to reside in main memory. In this configuration, the performance bottleneck is likely to be the gap between the processing speed of the CPU and the memory access latency. Previous work has shown that database applications have large

instruction and data footprints and hence do not use processor caches effectively. In this paper, we propose Call Graph Prefetching (CGP), ...

Keywords: Instruction cache prefetching, call graph, database

20 Wrong-path instruction prefetching

Jim Pierce, Trevor Mudge

December 1996 **Proceedings of the 29th annual ACM/IEEE international symposium on Microarchitecture**

Full text available: pdf(1.25 MB)   Additional Information: full citation, abstract, references, citings, index terms

Instruction cache misses can severely limit the performance of both superscalar processors and high speed sequential machines. Instruction prefetch algorithms attempt to reduce the performance degradation by bringing lines into the instruction cache before they are needed by the CPU fetch unit. There have been several algorithms proposed to do this, most notably next line prefetching and target prefetching. We propose a new scheme called wrong-path prefetching which combines next-line prefetchin ...

Results 1 - 20 of 200   Result page: **1**   2   3   4   5   6   7   8   9   10   next

ⓐ P⌖RTAL
USPTO

THE GUIDE TO COMPUTING LITERATURE

ℹ Feedback Report a problem Satisfaction survey

Terms used **impact prefetch**   Found **2,778** of 873,144

Sort results by   [relevance ▽]   ❖ Save results to a Binder   Try an Advanced Search
Display results   [expanded form ▽]   ⁇ Search Tips   Try this search in The Digital Library
                                           ☐ Open results in a new window

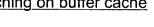Results 1 - 20 of 200   Result page: **1** 2 3 4 5 6 7 8 9 10   next
Best 200 shown   Relevance scale ☐ ▢ ▢ ■ ■

1 Caching & file systems: The performance impact of kernel prefetching on buffer cache replacement algorithms ■
Ali R. Butt, Chris Gniady, Y. Charlie Hu
June 2005 **Proceedings of the 2005 ACM SIGMETRICS international conference on Measurement and modeling of computer systems**
Full text available: 📄 pdf(303.47 KB)   Additional Information: full citation, abstract, references, index terms

> A fundamental challenge in improving the file system performance is to design effective block replacement algorithms to minimize buffer cache misses. Despite the well-known interactions between prefetching and caching, almost all buffer cache replacement algorithms have been proposed and studied comparatively without taking into account file system prefetching which exists in all modern operating systems. This paper shows that such kernel prefetching can have a significant impact on the relative ...

> **Keywords**: buffer caching, prefetching, replacement algorithms

2 Architectural and compiler support for effective instruction prefetching: a cooperative approach ■
February 2001 **ACM Transactions on Computer Systems (TOCS)**, Volume 19 Issue 1
Full text available: 📄 pdf(432.96 KB)   Additional Information: full citation, abstract, references, citings, index terms, review

> Instruction cache miss latency is becoming an increasingly important performance bottleneck, especially for commercial applications. Although instruction prefetching is an attractive technique for tolerating this latency, we find that existing prefetching schemes are insufficient for modern superscalar processors, since they fail to issue prefetches early enough (particularly for nonsequential accesses). To overcome these limitations, we propose a new instruction prefetching technique where ...

> **Keywords**: compiler optimization, instruction prefetching

3 The interaction of software prefetching with ILP processors in shared-memory systems ■
Parthasarathy Ranganathan, Vijay S. Pai, Hazim Abdel-Shafi, Sarita V. Adve
May 1997 **ACM SIGARCH Computer Architecture News , Proceedings of the 24th annual international symposium on Computer architecture**, Volume 25 Issue 2
Additional Information: full citation, abstract, references, citings, index

Full text available: pdf(2.44 MB)   terms

Current microprocessors aggressively exploit instruction-level parallelism (ILP) through techniques such as multiple issue, dynamic scheduling, and non-blocking reads. Recent work has shown that memory latency remains a significant performance bottleneck for shared-memory multiprocessor systems built of such processors.This paper provides the first study of the effectiveness of software-controlled non-binding prefetching in shared memory multiprocessors built of state-of-the-art ILP-based proces ...

4 Reducing the impact of software prefetching on register pressure

David W. Shrewsbury, Cindy Norris
March 2000 **Proceedings of the 2000 ACM symposium on Applied computing - Volume 2**

Full text available: pdf(492.51 KB)   Additional Information: full citation, references, citings, index terms

5 Performance comparison of thrashing control policies for concurrent Mergesorts with parallel prefetching

Kun-Lung Wu, Philip S. Yu, James Z. Teng
June 1993 **ACM SIGMETRICS Performance Evaluation Review , Proceedings of the 1993 ACM SIGMETRICS conference on Measurement and modeling of computer systems**, Volume 21 Issue 1

Full text available: pdf(1.23 MB)   Additional Information: full citation, abstract, references, citings, index terms

We study the performance of various run-time thrashing control policies for the merge phase of concurrent mergesorts using parallel prefetching, where initial sorted runs are stored on multiple disks and the final sorted run is written back to another dedicated disk. Parallel prefetching via multiple disks can be attractive in reducing the response times for concurrent mergesorts. However, severe *thrashing* may develop due to imbalances between input and output rates, thus a large number o ...

6 Evaluating the impact of memory system performance on software prefetching and locality optimizations

Abdel-Hameed A. Badawy, Aneesh Aggarwal, Donald Yeung, Chau-Wen Tseng
June 2001 **Proceedings of the 15th international conference on Supercomputing**

Full text available: pdf(387.06 KB)   Additional Information: full citation, abstract, references, citings, index terms

Software prefetching and locality optimizations are techniques for overcoming the speed gap between processor and memory. In this paper, we evaluate the impact of memory trends on the effectiveness of software prefetching and locality optimizations for three types of applications: regular scientific codes, irregular scientific codes, and pointer-chasing codes. We find for many applications, software prefetching outperforms locality optimizations when there is sufficient memory bandwidth, but ...

7 Cooperative prefetching: compiler and hardware support for effective instruction prefetching in modern processors

Chi-Keung Luk, Todd C. Mowry
November 1998 **Proceedings of the 31st annual ACM/IEEE international symposium on Microarchitecture**

Full text available: pdf(3.41 MB)   Additional Information: full citation, references, citings, index terms

8

A general framework for prefetch scheduling in linked data structures and its

application to multi-chain prefetching
Seungryul Choi, Nicholas Kohout, Sumit Pamnani, Dongkeun Kim, Donald Yeung
May 2004 **ACM Transactions on Computer Systems (TOCS)**, Volume 22 Issue 2

Full text available: pdf(2.45 MB)   Additional Information: full citation, abstract, references, index terms

Pointer-chasing applications tend to traverse composite data structures consisting of
multiple independent pointer chains. While the traversal of any single pointer chain leads to
the serialization of memory operations, the traversal of independent pointer chains provides
a source of memory parallelism. This article investigates exploiting such *interchain memory
parallelism* for the purpose of memory latency tolerance, using a technique called *multi--
chain prefetching*. Previous work ...

Keywords: Data prefetching, memory parallelism, pointer-chasing code

9 Tolerating latency in multiprocessors through compiler-inserted prefetching
Todd C. Mowry
February 1998 **ACM Transactions on Computer Systems (TOCS)**, Volume 16 Issue 1

Full text available: pdf(410.70 KB)   Additional Information: full citation, abstract, references, citings, index
                                                                        terms, review

The large latency of memory accesses in large-scale shared-memory multiprocessors is a
key obstacle to achieving high processor utilization. Software-controlled prefetching is a
technique for tolerating memory latency by explicitly executing instructions to move data
close to the processor before the data are actually needed. To minimize the burden on the
programmer, compiler support is needed to automatically insert prefetch instructions into
the code. A key challenge when ...

Keywords: compiler optimization, prefetching

10 Speeding up irregular applications in shared-memory multiprocessors: memory binding
and group prefetching
Zheng Zhang, Josep Torrellas
May 1995 **ACM SIGARCH Computer Architecture News , Proceedings of the 22nd
              annual international symposium on Computer architecture**, Volume 23 Issue 2

Full text available: pdf(1.74 MB)   Additional Information: full citation, abstract, references, citings, index
                                                                        terms

While many parallel applications exhibit good spatial locality, other important codes in areas
like graph problem-solving or CAD do not. Often, these irregular codes contain small records
accessed via pointers. Consequently, while the former applications benefit from long cache
lines, the latter prefer short lines. One good solution is to combine short lines with
prefetching. In this way, each application can exploit the amount of spatial locality that it
has. However, prefetching, if provided, ...

11 The performance impact of I/O optimizations and disk improvements
W. Hsu, A. J. Smith
March 2004 **IBM Journal of Research and Development**, Volume 48 Issue 2

Additional Information: full citation, abstract, references, index terms

In this paper, we use real server and personal computer workloads to systematically analyze
the true performance impact of various I/O optimization techniques, including read caching,
sequential prefetching, opportunistic prefetching, write buffering, request scheduling,
striping, and short-stroking. We also break down disk technology improvement into four
basic effects--faster seeks, higher RPM, linear density improvement, and increase in track
density--and analyze each separately to determi ...

12 <u>Compiler-based I/O prefetching for out-of-core applications</u>
Angela Demke Brown, Todd C. Mowry, Orran Krieger
May 2001 **ACM Transactions on Computer Systems (TOCS)**, Volume 19 Issue 2

Full text available: pdf(499.03 KB)   Additional Information: <u>full citation</u>, <u>abstract</u>, <u>references</u>, <u>citings</u>, <u>index terms</u>, <u>review</u>

Current operating systems offer poor performance when a numeric application's working set does not fit in main memory. As a result, programmers who wish to solve "out-of-core" problems efficiently are typically faced with the onerous task of rewriting an application to use explicit I/O operations (e.g., read/write). In this paper, we propose and evaluate a fully automatic technique which liberates the programmer from this task, provides high performance, and requires only minima ...

Keywords: compiler optimization, prefetching, virtual memory

13 <u>Informed prefetching and caching</u>
R. H. Patterson, G. A. Gibson, E. Ginting, D. Stodolsky, J. Zelenka
December 1995 **ACM SIGOPS Operating Systems Review , Proceedings of the fifteenth ACM symposium on Operating systems principles**, Volume 29 Issue 5
Full text available: pdf(2.13 MB)   Additional Information: <u>full citation</u>, <u>references</u>, <u>citings</u>, <u>index terms</u>

14 <u>Prefetching for improved bus wrapper performance in cores</u>
Roman Lysecky, Frank Vahid
January 2002 **ACM Transactions on Design Automation of Electronic Systems (TODAES)**, Volume 7 Issue 1

Full text available: pdf(430.83 KB)   Additional Information: <u>full citation</u>, <u>abstract</u>, <u>references</u>, <u>citings</u>, <u>index terms</u>

Reuse of cores can reduce design time for systems-on-a-chip. Such reuse is dependent on being able to easily interface a core to any bus. To enable such interfacing, many propose separating a core's interface from its internals by using a bus wrapper. However, this separation can lead to a performance penalty when reading a core's internal registers. In this paper, we introduce prefetching, which is analogous to caching, as a technique to reduce or eliminate this performance penalty, involving a ...

Keywords: Bus wrapper, PVCI, VSIA, cores, design reuse, intellectual property, interfacing, on-chip bus, system-on-a-chip

15 <u>Predictive Prefetching on the Web and Its Potential Impact in the Wide Area</u>
Christos Bouras, Agisilaos Konidaris, Dionysios Kostoulas
June 2004 **World Wide Web**, Volume 7 Issue 2

Full text available: Publisher Site   Additional Information: <u>full citation</u>, <u>abstract</u>, <u>index terms</u>

The rapid increase of World Wide Web users and the development of services with high bandwidth requirements have caused the substantial increase of response times for users on the Internet. Web latency would be significantly reduced, if browser, proxy or Web server software could make predictions about the pages that a user is most likely to request next, while the user is viewing the current page, and prefetch their content.

In this paper we study Predictive Prefetching on a totally ne ...

Keywords: Web Prefetching, caching, prediction algorithms, trace based simulation

16 <u>Configuration prefetch for single context reconfigurable coprocessors</u>
Scott Hauck
March 1998 **Proceedings of the 1998 ACM/SIGDA sixth international symposium on Field programmable gate arrays**

Full text available: 📄 <u>pdf(1.14 MB)</u>   Additional Information: <u>full citation</u>, <u>abstract</u>, <u>references</u>, <u>citings</u>, <u>index terms</u>

Current reconfigurable systems suffer from a significant overhead due to the time it takes to reconfigure their hardware. In order to deal with this overhead, and increase the power of reconfigurable systems, it is important to develop hardware and software systems to reduce or eliminate this delay. In this paper we propose one technique for significantly reducing the reconfiguration latency: the prefetching of configurations. By loading a configuration into the reconfigurable logic ...

17 <u>Instruction prefetching of systems codes with layout optimized for reduced cache misses</u>
Chun Xia, Josep Torrellas
May 1996 **ACM SIGARCH Computer Architecture News , Proceedings of the 23rd annual international symposium on Computer architecture**, Volume 24 Issue 2

Full text available: 📄 <u>pdf(1.65 MB)</u>   Additional Information: <u>full citation</u>, <u>abstract</u>, <u>references</u>, <u>citings</u>, <u>index terms</u>

High-performing on-chip instruction caches are crucial to keep fast processors busy. Unfortunately, while on-chip caches are usually successful at intercepting instruction fetches in loop-intensive engineering codes, they are less able to do so in large systems codes. To improve the performance of the latter codes, the compiler can be used to lay out the code in memory for reduced cache conflicts. Interestingly, such an operation leaves the code in a state that can be exploited by a new type of ...

18 <u>The Performance of Runtime Data Cache Prefetching in a Dynamic Optimization System</u>
Jiwei Lu, Howard Chen, Rao Fu, Wei-Chung Hsu, Bobbie Othmer, Pen-Chung Yew, Dong-Yuan Chen
December 2003 **Proceedings of the 36th annual IEEE/ACM International Symposium on Microarchitecture**

Full text available: 📄 <u>pdf(253.79 KB)</u>   Additional Information: <u>full citation</u>, <u>abstract</u>, <u>citings</u>, <u>index terms</u>

Traditional software controlled data cache prefetching isoften ineffective due to the lack of runtime cache miss andmiss address information. To overcome this limitation, weimplement runtime data cache prefetching in the dynamicoptimization system ADORE (ADaptive Object code RE-optimization).Its performance has been compared withstatic software prefetching on the SPEC2000 benchmarksuite. Runtime cache prefetching shows better performance.On an Itanium 2 based Linux workstation, it can increasepe ...

19 <u>Limits on the performance benefits of multithreading and prefetching</u>
Beng-Hong Lim, Ricardo Bianchini
May 1996 **ACM SIGMETRICS Performance Evaluation Review , Proceedings of the 1996 ACM SIGMETRICS international conference on Measurement and modeling of computer systems**, Volume 24 Issue 1

Full text available: 📄 <u>pdf(1.17 MB)</u>   Additional Information: <u>full citation</u>, <u>abstract</u>, <u>references</u>, <u>citings</u>, <u>index terms</u>

This paper presents new analytical models of the performance benefits of multithreading and prefetching, and experimental measurements of parallel applications on the MIT Alewife

multiprocessor. For the first time, both techniques are evaluated on a real machine as opposed to simulations. The models determine the region in the parameter space where the techniques are most effective, while the measurements determine the region where the applications lie. We find that these regions do not always o ...

20 <u>Tolerating memory latency through push prefetching for pointer-intensive applications</u> ■
Chia-Lin Yang, Alvin R. Lebeck, Hung-Wei Tseng, Chien-Hao Lee
December 2004 **ACM Transactions on Architecture and Code Optimization (TACO)**, Volume
1 Issue 4
Full text available: 🗋 <u>pdf(590.24 KB)</u>   Additional Information: <u>full citation</u>, <u>abstract</u>, <u>references</u>, <u>index terms</u>

Prefetching is often used to overlap memory latency with computation for array-based applications. However, prefetching for pointer-intensive applications remains a challenge because of the irregular memory access pattern and pointer-chasing problem. In this paper, we proposed a cooperative hardware/software prefetching framework, the push architecture, which is designed specifically for linked data structures. The push architecture exploits program structure for future address generation instea ...

Keywords: Prefetch, linked data structures, memory hierarchy, pointer-chasing

Results 1 - 20 of 200   Result page: **1**   <u>2</u>   <u>3</u>   <u>4</u>   <u>5</u>   <u>6</u>   <u>7</u>   <u>8</u>   <u>9</u>   <u>10</u>   <u>next</u>